

Article

# BSM-Data Reuse Model Based on In-Vehicular Computing

Khiredine Benaissa <sup>1</sup>, Salim Bitam <sup>2</sup>  and Abdelhamid Mellouk <sup>3,4,\*</sup> 

<sup>1</sup> Department of Computer Science, University of Medea, Medea 26000, Algeria; benaissa.khiredine@univ-medea.dz

<sup>2</sup> LESIA Laboratory, Department of Computer Science, University of Biskra, Biskra 07000, Algeria; s.bitam@univ-biskra.dz

<sup>3</sup> University Paris-Est de Créteil, 94000 Créteil, France

<sup>4</sup> University of Oran 1, Oran 31000, Algeria

\* Correspondence: mellouk@u-pec.fr

Received: 5 July 2020; Accepted: 29 July 2020; Published: 6 August 2020



**Abstract:** Basic Safety Messages that are frequently generated from multiple connected vehicles can play a primordial role in providing transport data see credible and reliable information they contain. Otherwise, when considering the way Basic Safety Messages (BSMs) are treated, multiple deficiencies prevent the latter to be capable of constituting a precious data source. As we know, data become more useful the more widely are used, which is the exact opposite of what happens with the BSMs that exist only temporarily, used locally, considered disposable, and are never stored. In this paper, we introduce a data reuse model that retains collected BSMs, stores, and processes them inside the vehicle constituting a continuous data source holding retained snapshots along the roadway. Our model provided a primary data source available on a large scale, considered to be a worthy dataset for machine learning tasks, capable of visualizing different traffic-related indicators to enhance analytics and support decisions-making. In the study case, we set up an in-vehicle data platform, where we achieved an 80% of BSMs size reduction and provided a rich set of APIs to serve applications. We also adopted the Artificial Neural Networks (ANN) as an information processing paradigm for performing traffic volume prediction, where the obtained results have reached over 99% of accuracy.

**Keywords:** BSM; data reuse; in-vehicule computing; connected vehicle; VANETs; machine learning; ANN

## 1. Introduction

The increasing volume of traffic that cities currently face is associated with many unpleasant phenomena, such as accidents, time delays, emergencies, as well as high pollution and degradation of life quality. Nevertheless, with the digital age constantly moving forward, a revolution in the transportation network is being spurred by advancements in communication technologies. Therefore, governments, academia, and industrials making tremendous efforts have made advancement to reinforce the evolution of the commonly named intelligent transportation system (ITS). Meanwhile, cars are getting increasingly intelligent. A contemporary car comprises over 20,000 components, about 40 microprocessors and dozens of sensors. In addition, an eclectic selection of technologies strive to offer different vehicular communications models known as vehicle-to-everything (V2X). These progresses in sensing technologies are inaugurating new possibilities, such as connected vehicles (CV). As one of the most heavily researched automotive technologies, CV technology aims at introducing improvements concerning safety and efficiency of the transportation system and roads. It

then has the possibility to ameliorate ongoing activities, modify transportation system management, complete or substitute traditional data collection approaches, and essentially amend the paradigm of the transportation data. Hence, it is not surprising that "data" is the most common word that appeared in all responses in a request for information that concerned connected vehicle generalization project in the U.S. (<https://www.federalregister.gov/documents/2014/03/12/2014-05414/connected-vehicle-pilot-deployment-program-request-for-information>).

Next to that, the blast of transport data and its assured worth is pushing present transportation models to change at a rapid pace; a transformative change that not only evaluates the importance of data, but also how efficiently it can be processed near the source of generation. According to Gartner, a leading IT research company, "predicts that by 2025, 75% of data will be processed outside the centralized data center or cloud" (<https://www.gartner.com/smarterwithgartner/what-edge-computing-means-for-infrastructure-and-operations-leaders/>). The company has also announced the edge computing as one of its top ten strategic technology trends for 2020, when it says, "Edge computing will become a dominant factor across virtually all industries and use cases as the edge is empowered with increasingly more sophisticated and specialized compute resources and more data storage" (<https://www.forbes.com/sites/peterhigh/2019/10/21/breaking-gartner-announces-top-10-strategic-technology-trends-for-2020/#6da7af840744>). As presented in Figure 1 from Premio (<https://premioinc.com/pages/in-vehicle-fanless-computer>), today's vehicle computing solutions propose complete models for all requirements of in-vehicle computing and offer industrial-grade in-vehicle computers.



**Figure 1.** In-vehicle-fanless-computer.

Consequently, intelligent transportation systems are required in order to shift to mobile computing technology for the flexibility that they need to meet the daily challenges of demanding connected and autonomous vehicles applications.

Dedicated Short Range Communication (DSRC) protocol has been adopted as a leading communication standard in the U.S., for establishing an advanced connected vehicle network and supporting traffic efficiency and safety, along with SAE J2735 as a primary standard to provide V2X data guidelines for deploying applications for DSRC-enabled connected vehicle. Actually, the DSRC and SAE-J2735 together grant the potential to exchange safety data wirelessly and very effectively using what is presented as basic safety messages (BSM) [1–3]. This commonly referred to as the "heartbeat" message, is nominated as the main data set used to continuously exchange safety data between connected vehicles. Over real-time exchange of this safety message, CV applications are able to discover and determine perilous situations. They can then inform drivers of the expected danger before it occurs or automatically acts toward impending incidents using vehicle control systems. Historically speaking, the primary purpose of the U.S. CV program was to enhance road safety by aiding vehicle-to-vehicle safety applications making the right decisions. Nevertheless, generating

data for transport agencies while using connected vehicles is, by design, a very important secondary principle. A great deal of the ongoing research under the mentioned program is oriented towards such data-providing applications. In this respect, BSMs frequently generated from multiple connected vehicles can play a primordial role in providing transport data see credible and reliable information that they contain. Otherwise, when considering the way BSMs are treated, multiple deficiencies prevent the latter to be capable of constituting a precious connected vehicle data source.

First: presently, the basic safety messages are particularly even exclusively used for the safety purpose. In contrast, a big bundle of its data elements are necessary for a considerable collection of applications not related to safety. At least, the BSM Part 1 data elements can conveniently provide basic vehicle information required by several applications. Arguably helpful, safety, mobility, and efficiency are not a separated aspects. Obviously, traffic accidents can breed Traffic congestion and increase CO<sub>2</sub> emission.

Next: BSMs as safety data are overall regarded as snapshot data that give an idea of the state of the system at a definite time. From the first use, the BSM becomes outdated, useless, and is deleted. Undoubtedly, BSMs collected from multiple connected vehicles can be cached, grouped and stored to construct continuous data streams that can supply almost real-time metrics while they evolve over space and time. Indeed, data are then exploited in a better-connected form to improve safety, decrease fuel consumption, reduce traffic jams, as well as facilitate people's travel overall.

Finally: as is well known, safety applications between vehicles require local broadcast of BSM within the limits of the DSRC. Even though, connected vehicles using applications are required to enable an extra-vehicular data exchange to permit collaborative sensing and action at scale. Thus, if transmitted according to different technologies other than DSRC such as Cellular V2X, continuous data obtained through BSMs processing can widely feed applications with requested vehicle information [4].

In our previous works, first, in [5] we proposed a Request-To-Receive (RTR) approach to address the blinding broadcast of BSM, to reduce bandwidth over-utilization, and decrease data loss and delay. Next, in [6] we proposed an Application-Level Flextime Aggregation (ALFA) scheme that caches BSMs on-board the vehicle and then aggregates them into new messages containing stored snapshots according to contexts for every stretch of roadway. In this paper, for broad, intelligent, and rational using of safety messages, we propose a BSM reuse model taking into account the general context of ITS, where connected vehicles are expected to boost safety, decrease traffic congestions, and reduce CO<sub>2</sub> emissions and fuel consumption [7].

Dealing BSM as wealthy data outside its baseline design, out of safety context and beyond DSRC limits has never been considered before. Our proposed model retains collected BSMs, stores, and processes them inside the vehicle to provide a continuous data source holding saved snapshots for each roadway segment.

The main innovation of our model is its participation to change the way that BSM is treated, opening up new possibilities to broadly reuse stored BSMs, making them more valued, and resulting in the creation of a reliable and useful transport data source. Moreover, given that our model relies on In-Vehicle Computing (IVC) paradigm, its strongest potential is, therefore, its capability of facilitates data processing at, or close to, their source of generation while optimizing bandwidth and reducing latency to rapidly provide data of value to drivers, pedestrians, and transportation agencies.

The rest of this article is structured, as follows: Section 2 shows brief overview of the methods and techniques for connected vehicle data collection and sharing in the existing literature. Section 3 describes the availability of data in consideration of connected vehicle technology. Section 4 introduces and describes our proposed data reuse model. Section 5 presents study case, data description, and results and discussions. Finally, our conclusions, key study limitations, and future work are presented in Section 6.

## 2. Related Works

If connected vehicles are to form the center of future intelligent transportation systems, one of the most glamorous areas that presents many challenges as we progress is transport data sources. There is no doubt that data have great potential and power, and data are the blood that will run through the veins of connected vehicles [8]. Connected vehicles' data can provide the ability to meet the modern interests of drivers like for instance "What is the best fuel-efficient route" and go beyond the classical inquiries about shortest-distance or earliest arrival route [9]. In this respect, the increasing focus of the ITS research community towards the topic of data in connected vehicle environment has led to a bustle of works introducing different approaches to the problem of data collection.

Indeed, data collection that is based on the recruitment of connected vehicles is recently studied of [10], where an optimal set of vehicles are required to be identified and recruited to carry out an urban data collection in favor of service provider to facilitate users on city-streets. This incentive-based method named REVERS has exploited game-theory to fairly and optimally select the best vehicles under desired coverage, redundancy, and quality requirements. Similarly, to perform urban sensing for a desired coverage, given a limited budget, the authors of [11] proposed the recruitment of high reputation users. Additionally, the authors of [12] have focused on how to maximize message coverage in urban vehicular networks by the optimal deployment of roadside units. In their proposed message coverage maximization algorithm, namely MCMA, the authors have considered traffic stream and delay constraint of applications to attentively identify the appropriate site for RSUs. Some other studies have been conducted on connected vehicles data transmission issues. Accordingly, the authors of [13] have attempted to improve the cooperative data dissemination performance, where the authors considered the cooperative data scheduling decision in hybrid (V2I) and (V2V) vehicle networks as a maximum weighted independent set (MWIS) problem. Subsequently, by solving the MWIS, the goal is to maximize the number of vehicles that retrieve their requested data. In the same way, the work by [14] has dealt with connected vehicles related data distribution tasks, where the authors designed an infrastructure to enable large-scale message delivery by labeling and customizing unstructured data into topics in order to serve a wide range of consumers. Authors of [15] have attempted to cope with the problem of delay-tolerant connected vehicles data traffic delivery. They introduced an architecture where delay-tolerant traffic is offloaded from the data networks to the connected vehicle networks, without extra infrastructure or hardware deployment. The authors have also proposed a distributed data hopping mechanism to allow delay-tolerant data routing over CV networks. In another paper [16], for large-scale vehicular content distribution in urban areas, the authors came up with the idea of deploying a multitude of wireless buffer devices on the roadside, namely roadside buffers (RSBs), to extend the distribution of locally content to vehicles in the urban area. The work by [17] discussed the possibility of substituting RSU by city buses widely distributed in the city area to improve intra-cluster BSM dissemination. The authors have also proposed an allocation mechanism for intra-cluster message distribution. When considering the problem of vehicular datasets, authors of [18] mimicked the ordinary daily road activity of a 400 km<sup>2</sup> region and generated a realistic artificial vehicular mobility dataset.

Regarding aforementioned studies, it is important to note that data collection approaches remain ineffective in a dynamic environment, since they are not based on stable criteria for the selection of candidate vehicles that should be important and available to meet the relevant user interests in the network. In addition, they are not scalable data collection for a given coverage and budget constraints [11,12]. Despite that the authors of [10] have adopted the Information-Centric Networking (ICN) concept, in their work, the information importance remains location-based, depends on the content popularity and computed through the observation of the amount and frequency of user interests it received, where they neglected whether the receiver is satisfied or not with this content. Moreover, there is no proper metric to classify and identify the participants' eligibility that has been used by the authors of [11]. In [12], the proposed deployment strategy can notably be influenced by large-scale mobility model and road layout.

In regards to data transmission works, the centralized model presented in [13] is limited to single-hop V2V communication and do not match by the SDN concept of “logically centralized” control in a distributed network. In [14], although the authors have respected the recommended latency requirements for CV applications, they considered in their experimentation latency that only includes the delivery time of a message and not the processing times, such as the time required for aggregation and complex data transformation. The proposed infrastructure presented in [16] is not extendable to any network size, so it is not capable to support Internet-enabled content distribution. Though the mobility model and road vehicle density can deadly influence the intra-cluster message dissemination, they have not been considered in the clustering process in [17]. The authors of [18] have described the generation process and outlined the impact of the generated dataset on the simulative evaluation of vehicular networks. Nevertheless, a real-world dataset remains necessary to allow for a more rigorous validation of the mobility models.

On the data side, not all previous work used realistic, reliable, and large-scale data sources, such as BSMs; each system instead has defined its proprietary data formats and collected its required data to provide its services. Consequently, different applications in these systems are not able to complement each other by reuse and sharing data, as they are unable to understand each other’s data. This hinders cross-application data reuse and optimizations (e.g., to reduce data traffic by reducing redundancy).

On the simulation side, some works were concerned by a local simulation area [11,13,15–17], a limited sample [14], and an unknown number [10] of simulated vehicles with limited buffer size [10,15]. The communication characteristics were not specified in [10,11,16,18] and simulated based only on DSRC technology in [12–14].

Besides, advances in vehicular communications technology are making content sharing within vehicular networks more effective and increasingly more popular [19]. Some other works have explored the potential of interworking between DSRC and cellular network technologies for efficient V2X communications in favor of data sharing. The work presented by [20] has investigated the possibility of leveraging DSRC and cellular interworking for successful V2X transmissions and examined possible DSRC and cellular combined architectures. Another study in [21] has taken advantage of the potential of V2X communications based on LTE technology to introduce a device-to-device content-sharing approach. V2V and V2I link planning take into account both data diversity and link quality.

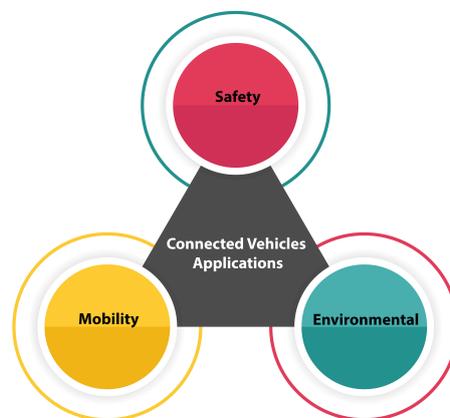
As can be seen, notwithstanding that connected vehicle data have recently got in the limelight of the research community, many of the previous and on-going researches have focused on the concept of data collection and sharing, yet the reuse of data has not been in focus. Notably, BSM reuse, which is of salient importance, has been disregarded in previous researches. In addition, the realities of data reuse are not yet straightforward. Some of the fundamental issues are technical, from identifying what datasets are available and the size of that data, to the wireless connections suitable for transmission.

In fact, not only BSM is particularly used for the safety purpose, but it becomes outdated, useless, and deleted From the first use. However, on the one hand, arguably helpful, safety, mobility, and efficiency are not a separated aspects. Obviously, traffic accidents can breed traffic congestion and increase CO<sub>2</sub> emission. On the other hand, although BSM was initially limited to the safety purpose, it has the possibility to be used outside safety. If BSMs data collected from multiple connected vehicles are be cached, grouped, stored and widely diffused, they surely help boosting safety, decreasing fuel consumption, reducing traffic jams, as well as facilitating people’s travel overall [6]. By way of illustration, a BSM reporting a risk is locally used by safety application. However, at the broadest level, the same BSM can be seen by efficiency applications as an input to estimate alternative paths.

Conversely, as far as we know, our work is the first to consider the reuse of BSM, which is a structured dataset to provide reliable, useful, and wide-scale data sources for connected vehicles. This BSMs reuse approach can provide data source to a variety of data-related services that support multi-modal transportation applications, not only for the safety purpose, but also efficiency and mobility.

### 3. Connected Vehicles and Data Availability

Connected-vehicles solution is of the many technological innovations currently jostling for attention. This new technology has revolutionized the automotive industries, and built the cornerstone of the internet of vehicles. According to [22], in 2020, the internet will be integrated in around 90% of modern cars, whereas it was integrated at less than 10% in 2013 and that certainly can help supporting next generation intelligent transportation systems. Vast research work and various industrial efforts have accelerated the achievement of connected vehicle technology. Different countries, such as the U.S. (in California, New York, Arizona, Florida, and Michigan), China, Germany, the U.K., and others, have established connected vehicles testbeds and pilot programs [23,24]. That is why the latter is commonly regarded as an area of development where applications find prosperous ground in the IoV epoch. By way of example, Figure 2, summarized the main categories of connected vehicle applications in the U.S.



**Figure 2.** Connected Vehicles applications.

Indeed, great endeavors are being made by researchers towards innovative and cost-effective vehicular applications. Moreover, several applications that are proposed or under investigation are mainly related to safety, mobility, efficiency and infotainment, similar to emergency warning, traffic management, and weather information. The two most important aspects for connected vehicles to succeed technologically are then, first, numerous data need to be collected from diverse systems and sources. Second, these data should be treated and widely diffused through various communication technologies, such as DSRC, WiFi, 5G, and cellular.

Against this background, the general understanding is that better availability of data source is of the utmost importance to feed the plethora of connected vehicle applications and provide an intelligent transportation system management.

As the main thrust behind the connected vehicles is traffic safety, the BSM was initially designed to be the main message used by safety applications to share data among connected vehicles. Being considered as “heartbeat” messages, the BSM data tend to constitute the overwhelming majority of the CV data. Unfortunately, this valuable CV data source remains restricted in context, time and space.

#### 3.1. Vehicular Data Representation

Vehicles are getting more and more intelligent. An average car today contains more than 20,000 components, including about 40 microprocessors and an important set of embedded sensors that can number up to 200 sensors per vehicle in 2020 [25]. Modern vehicles hinge on these considerable sets of sensors in order to generate and exchange vehicle motion and status data. Thereby, in a connected vehicle scenario, a rich data source is the vehicle itself. Nevertheless, most data that are generated

by a vehicle are primarily of a technical nature; differ from carmaker to carmaker, and even within carmakers, from model to model.

When considering the connected vehicles’ technology, which aims at sharing some of these data with third parties, a variety of data representation known as messages sets have been proposed to support interoperability and enable data exchanges among connected vehicles network. The Society of Automotive Engineers (SAE) has developed the J2735 standard, which specifies a Message Set Dictionary, explicitly to support interoperability among applications based on the DSRC [26]. The SAE standard J2735 defines approximately 150 standard data elements and 70 standard data frames and describes 15 types of application data messages sets listed in Table 1.

**Table 1.** Society of Automotive Engineers (SAE) J2735 Dedicated Short Range Communication (DSRC) standard message sets.

Message Set	Purpose
A La, Carte Message	Generic message with flexible content.
Basic Safety Message	Transmits information necessary for vehicle-to-vehicle safety applications
Common Safety Request	A vehicle uses this to request specific state information from another vehicle
Emergency Vehicle Alert Message	Alerts, drivers that an emergency vehicle is active in an area
Intersection Collision Avoidance	Provides, vehicle location information relative to a specific intersection
Map Data	A roadside unit uses this to convey the geographic description of an,intersection
NMEA, Corrections	Encapsulates, one style of GPS corrections—NMEA (National Marine Electronics Association),style 183
Probe, Data Management	A, roadside unit uses this to manage the collection of probe data from vehicles
Probe, Vehicle Data	Vehicles, report their status over a given section of road to allow a roadside unit to, derive road and traffic conditions

A message is a combination between two structures named data frame and data element. The data frame is a complex data structure that contains one or more data elements and even other data frames. As stated in [27], among the fifteen messages described in the J2735, the BSM is considered to be the more important.

### 3.2. Basic Safety Message Data

Connected vehicle safety application are greatly dependent BSM to exchange the core data that describe vehicle status, position, and motion among vehicles, as well as between vehicle and Infrastructure. The BSM has been designed with two parts (see Figure 3, for the format of the BSM). The Part I contains the core data information and is transmitted regularly. The second part consists of other data elements that differ according to the vehicle model.

Table 2 groups Part I data elements. This content presents the official data element and data frame terminology from the standard. The Acceleration Set4Way and VehicleSize items are based on data frames, and the remaining items are based on data elements [27].

In this paper, we focus on study and consideration of BSM as an original and affluent data source. BSMs that contain position and motion data, state information of the vehicles (e.g., latitude, longitude, elevation, heading, speed, acceleration, lights, brakes, wipers, time-stamped, path history) exist only temporarily, used locally, and are never stored.

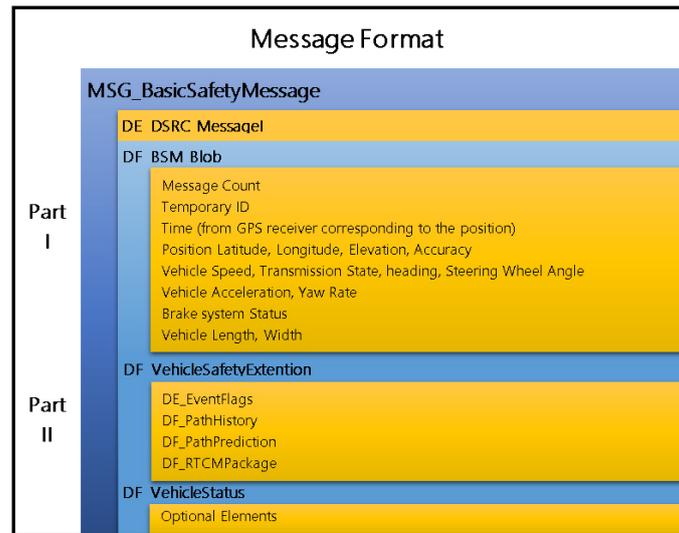


Figure 3. SAE J2735 Basic Safety Message Format.

Table 2. SAE J2735 basic safety message, Part I.

Num	Designation
1	DSRCmsgID
2	MsgCount
3	Common Safety Request
4	TemporaryID
5	DSecond
6	Latitude
7	Positional Accuracy
8	Transmission & Speed
9	Heading
10	Steering Wheel Angle
11	AccelerationSet 4Way
12	Brake System Status
13	Vehicle Size

#### 4. BSM Reuse Model: Our Proposal

##### 4.1. Problem Specification

BSMs frequently generated from multiple connected vehicles can play a primordial role in providing transport data and see the credible and reliable information they contain. Otherwise, given the way that BSMs are considered and treated, multiple deficiencies prevent the latter to be capable of constituting a precious connected vehicle data source. Therefore, the following features can be listed.

Context: presently, the basic safety messages are particularly even exclusively used for the safety purpose. In contrast, a big bundle of its data elements are necessary for a considerable collection of applications not related to safety. At least, the BSM Part 1 data elements can conveniently provide basic vehicle information required by several applications. Arguably helpful, safety, mobility, and efficiency are not a separated aspects. Obviously, traffic accidents can breed Traffic congestion and increase CO<sub>2</sub> emission.

Validity: BSMs as safety data are overall regarded as snapshot data that give an idea of the state of the system at a definite time. From the first use, the BSM becomes outdated, useless, and is deleted.

Undoubtedly, BSMs collected from multiple connected vehicles can be cached, grouped and stored to construct continuous data streams that can supply almost real-time metrics, while they evolve over space and time. Indeed, data are then exploited in a better-connected form to improve safety, decrease fuel consumption, reduce traffic jams, as well as facilitate people's travel overall.

Range: as is well known, safety applications between vehicles require local broadcast of BSM within the limits of the DSRC. Even though, connected vehicles using applications are required in order to enable an extra-vehicular data exchange to permit collaborative sensing and action at scale. Thus, if transmitted according to different technologies other than DSRC, such as Cellular V2X, continuous data obtained through BSMs processing can widely feed applications with requested vehicle information.

#### 4.2. Basic Concepts

To overcome the aforementioned difficulties, we propose a new BSM reuse model that makes use of a three-stages life cycle process. The new model is described below and graphically illustrated in Figure 4.



Figure 4. Proposed Basic Safety Messages (BSM) life cycle.

The model that is shown in Figure 4 aims to represent a view of the different stages of the BSM life cycle. Given our primary goal of not constraining BSM use contextually and geographically, the main idea is that no BSM would be destroyed; all captured BSMs should be maintained, processed and reused in different ways to create value from BSMs and deal with them as wealthy data source outside their baseline design, out of safety context and beyond DSRC.

##### 4.2.1. Data Capture

The data capture stage refers to two different parts, the generation and the acquisition of data. The data acquisition serves as the collection of extra-vehicular data in the form of BSM as well as data generation represents the creation of intra-vehicular data through local sensor observations. An additional real-time pre-processing task is also to be conducted in order to classify and filter captured data.

##### 4.2.2. Data Maintain and Processing

When it comes to the maintaining and processing stage, a series of actions would be performed on raw BSMs to model, clean, compress, aggregate, organize, store, and extract data in an appropriate output form for subsequent use.

##### 4.2.3. Data Reuse

The last stage aims at opening up new possibilities for endless reuse of stored BSMs. In actual fact, several data consumers use cases may require different data delivery types. For example, a safety application or an emergency vehicle service may require a real-time dataset when an accident takes place. Contrariwise, a data analytics company might opt for historical car data in order to understand

traffic trends. This stage relies on different data delivery and visualization methods to cater to these different use case requirements as well as for the purpose of knowledge production.

#### 4.3. In-Vehicle Computing: Advantages

Vehicles are getting more intelligent and well equipped. Emerging intelligent vehicles will possess sufficient storage and computing resources to perform tasks locally, thus reducing the network load and delays. Contemporary vehicles are capable of owning a computer inside, which is an industrial edge-computer that is designed to sustain the rigors of vehicular environments while capturing, storing, and analyzing data from various sensors and devices required for Intelligent Transportation System applications. In-Vehicle Computing will then become paramount to substitute the classical Vehicle Cloud Computing (VCC). Table 3 highlights the differences between IVC and VCC according to different features. In addition, the key advantages of IVC can be summarized, as follows.

- A Storage: ncontrary to the centralized topology, the IVC permits the data storage inside the vehicle in the vicinity of their source of generation. This provides timely access to stored data and decreases the remot storage load.
- B Bandwidth: in the era of the connected vehicle, the amount of generated data is growing explosively and the content demands will further become varied. When considering the distance from users in centralized topology, cloud computing cannot assure the bandwidth requirements for delivering and remote processing of such a large amount of data. By mounting the computation and storage resources on vehicles, IVC is able to properly mitigate the high-bandwidth pressure.
- C Response Time: processing time with delivery time together represent the response time. In the case of centralized topology, the response time is considerable due to the delivery delay. In our decentralized topology IVC, the mounted computer as processing units is inside vehicle. Thus, the took responding time is significantly less, which enables connected vehicles to respond with more efficiency, better service, and further innovation through new applications.
- D Contextual data: in decentralized topology, users are able to obtain real-time information related to the behavior and location of vehicles, traffic conditions, network environment, etc. Accordingly, different applications would be improved. For instance, real-time information can be delivered to various vehicular users in accordance with their interests.

**Table 3.** Comparison between In-Vehicle Computing (IVC) and Vehicle Cloud Computing (VCC).

Features	In-Vehicle Computing	Vehicle Cloud Computing
Latency	Very Low	High
Deployment cost	Low	High
Communication	Real Time	Bandwidth Constrained
Location	Close proximity to end-user	Remote location
Decision making	Local	Centralized and Remote
Burden on core network	Low	High
Computation Capability	Medium	High
Storage Capacity	Limited	Highly Scalable
Mobility support	High	Limited

In accordance with the foregoing and taking advantage of the emerging Vehicle-mounted computing technology, we propose our IVC-based model, which relies on data storage, and processing inside the vehicle and that can definitely help to address the costs of bandwidth and enable more efficient real-time applications that require fast processing and response.

#### 4.4. Architecture Design

In our architecture, an in-vehicle computer serves as an edge computer and permits data to be processed and stored close to its source. The captured BSMs do not need to travel across the roads to a central data center, as it would in a traditional cloud-based architecture; nevertheless, the speed remains considerably faster, maintaining the latency much low.

As illustrated in Figure 5, a vehicle performs the aforementioned three-stages, as follows.

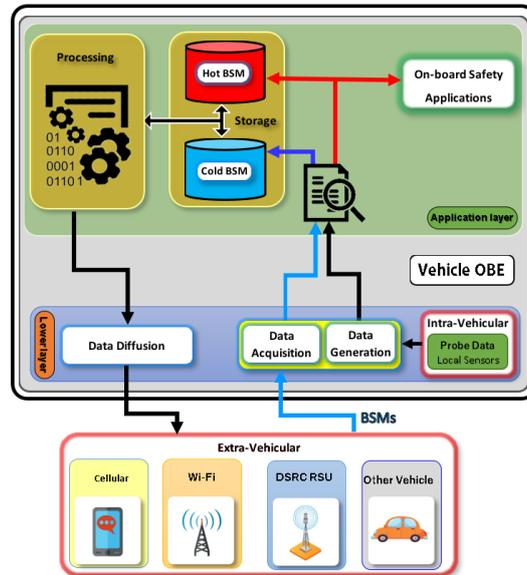


Figure 5. Basic architectural concept of our BSMs reuse model.

##### 4.4.1. Data Capture

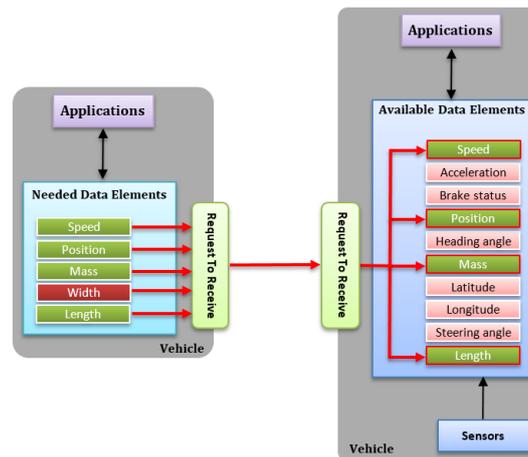
At this first stage, to manage the data transfer process, we adopt our previous Request-To-Receive approach [5] showed in Figure 6, to address the blinding exchange of BSMs and reduces the average number of collected Data Element. Performing this type of validation early on has a positive impact on the bandwidth capacity.

- A Categorization: further, we introduce a new concept of ‘data temperature’ to categorize the raw data captured accordingly. Hot data represent real-time BSMs and necessitate real-time processing to be more beneficial (i.e., less than a second from receipt to action). Hot data are also cached in a database shown in Figure 5, with red color. Cold data denote offline BSMs and are stored in blue database. Flextime processes can operate with cold data that have been stored. Hot data are simultaneously delivered alone to safety applications and to the storage function along with offline data.
- B Filtration: without any type of filtration, vehicles could easily get flooded with data. Data filtration addresses the issue of uninformative content of received raw BSM. The non-informative content can be real-time detected just by checking whether received BSM holds new data or not. If it is the case, the BSM is stored, otherwise, it comes to non-informative BSM that will be discarded. For instance, if a vehicle travels at more-or-less the same speed and heading during the entire trajectory, the data will essentially unchanged, then there would most probably be no loss of information. Therefore, a proper two-steps algorithm is performed in order to fulfill the categorization and filtration tasks. The algorithm will save valuable information and discard the rest.

Step 1: the designed algorithm examines the data element DSecond to know whether the message is possessing a real-time or offline data. The period of the message transmission is supervised by the DSecond data element. The later provides a time value when a BSM is populated with data there may

be a lag between the time the data is collected and populated in the BSM. BSMs are then grouped into two categories: hot and cool. Each of these two groups of data will require different kinds of processing and storage functions.

Step 2: the algorithm inspects the value of two data elements TemporaryID and the MsgCount of every hot BSM. It discards the BSM having old content but simultaneously notify local safety applications and storage each BSM having new content.



**Figure 6.** The basic architectural concept of Request-To-Receive approach.

#### 4.4.2. Data Maintain and Processing

Maintaining the captured BSM is the main task in our approach focused on preparing data for analysis and further reuse. It refers to data storage, modeling, reduction, and aggregation.

A Data storage and modeling: Messages, Frames, or Elements? After capturing the raw data, there is a requirement to transmit the data to suitable in-vehicle data storage systems for further processing and reuse. Accordingly, consideration should be given to how data are stored. Referring to the data is received as messages. The messages contain frames and the frames contain data elements. Connected vehicles and other related system functions usually require the use of data elements, but each element needs a location in time to make it useful.

- Messages: storing the data as messages would demand any future use of this information to inquire and access data elements across multiple messages. Furthermore, these messages will contain multiple elements that are not used by the function and may accordingly be inefficient to be accessed, used, and/or transmitted.
- Frames: to store the data as frames alone would not suffice, since many of the data elements are not necessarily in a frame.
- Element: if data are stored as elements alone, then any information concerning the association between data elements is lost. For example, if a BSM is divided into its elements, relationships between windshield wipers activation and temperature in a particular vehicle would not be known. However, such associations between data contained in messages generated by an individual vehicle can be generally accommodated by using the temporary ID assigned to messages sets to associate data for access or use. Because the association between data with one vehicle can be accommodated, it may be beneficial to store the data as elements. Overall, storing BSMs as elements that are associated with each other would allow each function or request to obtain information to only access the data elements it requires. Accordingly, we propose the relationship modelization that is shown in Figure 7. This approach allows for elements of the same type from different messages to be grouped together. For example, if the vehicle is responding to an inquiry from other connected vehicle requesting weather data, it would be able to read elements such as temperature and

windshield wiper activations from a single query, even though the data had been retrieved from multiple messages.

B Data reduction: data reduction is responsible for decreasing data storage requirements and communications bandwidth. A range of data reduction techniques may be appropriate to allow our model to minimize data storage requirements and communications bandwidth.

1 Compression: BSMZip (lossless compression for basic safety messages) Multiple compression techniques, including both lossy and lossless, could be applied to connected vehicle data. Lossless compression ensures data integrity and is more suitable for BSMs data.

In our model, we consider the application of run-length encoding (RLE) on a stream of BSM data. To the best of our knowledge, it has not been applied yet for data handling in the automotive domain.

2 Aggregation: different data aggregation strategies are appropriate to allow our model to perform a wide range of data processing, summarization, and display. Our model contains complicated aggregations on particular data elements, geo-fences, and some parameters for particular routes and areas of interest for end-users. Some of the connected vehicle applications may need to adopt a geo-fencing technique to help with limiting the data to be exchanged. This aforementioned technique defines an area of interest by drawing a boundary on BSMs, inside this defined area, specific data processing function can be accomplished. For instance, a speed detector on a highway may be a rectangle covering all lanes, in which any BSMs may have this speed qualified for further processing. Moreover, contextual aggregation of stored BSMs is performed using our previous ALFA scheme [5] to open up new possibilities of using BSMs outside the safety context.

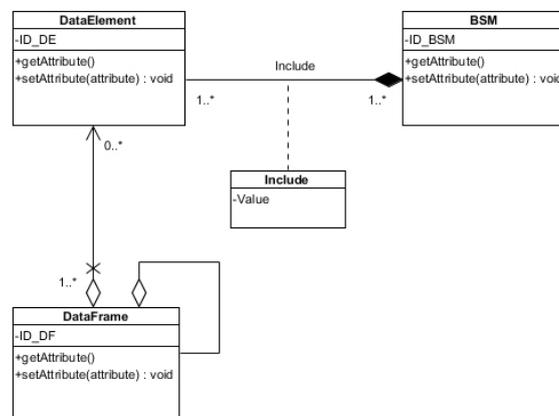


Figure 7. BSM modeling.

#### 4.4.3. Data Reuse

Next to the data maintaining function discussed above, access to data is a critical enabler for the efficient and wide reuse of BSMs into a multimodal transportation system. Hence, many data consumer use cases that may require different types of data delivery should be considered. Our model provides several data formats and it relies on different data delivery mechanisms to cater to these different use case requirements.

A Data reshaping formats: we can imagine and design a bunch of solutions such as a suite of APIs, portals, and apps to turn the passive stored BSMs into an active and actionable dataset and make every data element count. Hence, to support that and make this valuable dataset available for sharing and consumption, we count on a handful of data formats, as shown in Figure 8, including JSON, XML, and CSV in order to provide standard data that developers, systems, and

applications can easily reuse. Neither type is better than the other, we simply provide the ability for developers, systems, and applications to select the one that meets their requirements the best.

- B Data delivery: as we mentioned earlier, our model serves two types of vehicle data, hot data that refers to real-time data and cold data represented in offline data. Data delivery is a sort of service that allows for different transport agents to re-use the stored BSMs across the following methods.
- 1 Streaming: usually, hot data are better served using a push mechanism, which ensures minimal delay and packet loss. Besides, streaming is the ideal delivery mechanism for applications that require hot, rich, vehicle data. As a means to guarantee optimal and rational streaming, we count on our previous RTR approach that permits requesters to determine filters, like Data Element list, geo-fencing, and maximum latency, so they timely only get their desired data.
  - 2 Data Query: we rely on this retrieval technique to open up to different data consumers' the possibility of making requests on our database to obtain desired data. Data query is, therefore, a pull mechanism to provide hot or cold data by having data requests.
- C Data analysis: another way to reuse stored BSMs is by the application of emerging data analytic methods like machine learning. Machine-learning techniques can inspect our BSM dataset and make possible the patterns recognition (like real-time vehicle traffic, and driver behavior different road traffic conditions), decision-making, and/or future trends forecasting.
- D Data visualization: from another perspective, the BSM dataset can feed various data visualization tools and techniques to provide monitoring data about vehicles on the road and support decision-making to significantly improve the efficiency of transport system operations. For instance, it is practicable to visualize different data elements according to time and geofencing limits.

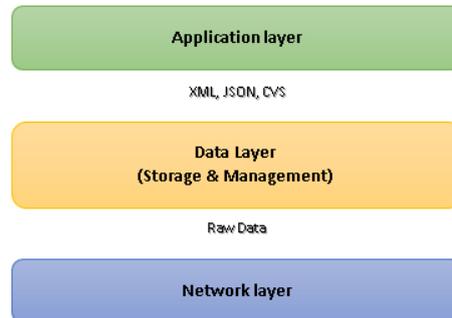


Figure 8. Data reuse formats.

## 5. Study Case: In-Vehicle BSM Data Services Platform

In the interest of delivering quick responses to end-users and enabling rapid storage and real-time data analysis, which is a vital feature for connected and autonomous vehicle applications, we carry out a real-world implementation of a vehicle data platform that is based on the reuse of collected and stored BSMs. Taking advantage of this available valuable data, In our study case, we only focus on the second and third stages of our proposed model. We also consider data reuse as a new other consumption of stored data in different cases through multiple scenarios.

### 5.1. Platform Implementation

- A Vehicle server: Hardware To implement our data services platform we mount a laptop as an in-vehicle computer along with a 4G LTE Wi-Fi Router. Table 4, shows the setting up hardware environment.
- B Vehicle server: Software Ensuring the different aforementioned functionalities may require different scenarios, which in turn demand different solutions. As well as, making the

development of such a data platform easier and faster pushes us to choose some suitable development frameworks. Choosing the wrong one may also severely affect our ability to design, develop, and maintain our software solution. As a result, adopting the design philosophy of Model-View-Controller (MVC), we set up a software development environment with Eclipse, Springboot, Maven, Tomcat, Postgres, MonetDB, Apache Kafka, MyBATIS, Anaconda, Tensorflow, Python, and Echart.

**Table 4.** The Hardware Environment Features.

<b>CPU</b>	Intel Core i7-9850H
<b>Memory</b>	32 GB
<b>Storage</b>	Internal 512GB PCIe SSD External 8TB SSD RAIDV3 USB3.1
<b>Graphics</b>	Intel UHD 630 with NVIDIA Q T2000 4GB
<b>WIRELESS</b>	LAN Dual Band Wi-Fi 6 AX200 IEEE 802.11ax
<b>4G LTE</b>	Wi-Fi Router 300Mbps (2.4GHZ) + 900Mbps (5GHz)

### 5.2. Data Preprocessing

A large amount of BSM data is accessible on the Safety Pilot Model Deployment Data (SPMD Project <https://catalog.data.gov/dataset/safety-pilot-model-deployment-data>), carried out in Ann Arbor, Michigan. The field test includes 75 miles of instrumented roadway. Approximately twenty-six roadside units (roadside equipment—RSE), which are capable of communicating with appropriately equipped vehicles, and devices via DSRC, were installed throughout the network, as presented in Figure 9.



**Figure 9.** Safety Pilot Model Deployment Site Plan, Ann Arbor, Michigan.

Approximately 3000 instrumented vehicles participated in this study. The vehicles include light/passenger vehicles, heavy/commercial trucks, and busses. We construct our relational database while using this rich and real-world connected vehicle dataset available under comma separated files (.csv) files. The result of our analysis on the available BSM file that is shown in Table 5, illustrates some of the summary measures that were populated with data collected on 11 April 2013.

**Table 5.** Summary Measures for Data Elements of the (.CSV) BSM File.

Field Name	No. of Unique Values	Min. Value	Max. Value	Sample Values	No. of Rows
BSMID	16,095,310			1,738,218,409, 1,801,843,621, 1,801,843,622, 1,920,703,252	16,095,310
DSRCMsgId	1			2	
MsgCount	128			108, 42, 0, 1, 2	
TemporaryID	11,587			−1,275,975,333, −1,157,424,605, −1,157,424,605, 738,663,065, 738,663,065,	
DSeconds	2194	0	65,535	0, 100, 200, 400, 59,900	
Latitude	456,480	0	900,000,001	423,091,009, 423,051,742, 423,072,124, 423,051,743	
Longitude	640,092	−840,921,953	180,000,001	−836,928,071, −836,847,010, −836,925,983, −836,925,983	
Elevation	1775	−12,773	7315	2510, 2436, 2401, 2436, 2511	
Positional Accuracy	5			0xE4FFFFFF, 0xE5FFFFFF, 0xFE000000, 0xFFFFFFFF	
Transmission State	5			0,1,2,3,7	
Speed	512	0	511	398, 177, 0, 381, 471	
Heading	28,805	−34	28,805	13,870, 28,708, 28,704, 5586	
SteeringWheelAngle	254	0	255	78, 144, 41, 47, 67	

### 5.2.1. Lossless Compression for BSM

Most of the data elements in this dataset are collected at a frequency of 10Hz. This frequency results in a number of the tables being very large, restricting the tables' ease of use. Looking to the "No. of Rows" column in Table 5, we can easily notice the huge amount of just a One Day generated data. Using the aforementioned dataset, we could observe that: most of the time, the majority of vehicles continue straight ahead at the same speed. As a result, if we take into account that a basic safety message is broadcast at 10 Hz, Most of the data elements will maintain their values during normal traffic flow. As shown in Table 6. If we consider a time period of two seconds with 20 messages, the only susceptible change for most of the two-second fragment of data is the position (longitude and latitude). Most often, no useful data are provided by messages 2 through 19. Despite that, it is crucial when a significant modification occurs in the data that the proper application receives it with minimum delay.

Effective opportunities for a significant reduction in storage and bandwidth requirements are evident in Table 6, yet any data element may also have individual compression techniques applied to it. In our work, we perform an individual compression on speed data element. Data compression methods, like run-length encoding, has exhibited its importance in such cases. However, traditional database systems (i.e., row-oriented databases) do not widely apply data compression techniques. On the contrary, column-oriented databases provide more opportunities for data compression as the values of the same attribute are stored consecutively [28]. Using columnar database like MonetDB, we found that RLE is an attractive approach for compressing data in a column-store. Run-length encoding compresses continuous duplicate values in a column to a compact singular representation. For example, RLE compresses  $k$  continuous duplicates whose value is  $t$  into one tuple  $(t; k)$ , i.e., (value, count) pair. RLE is widely used in column-oriented databases, where attributes are consecutively stored and runs of the same value are common.

**Table 6.** Illustration of Lossless compression for BSM.

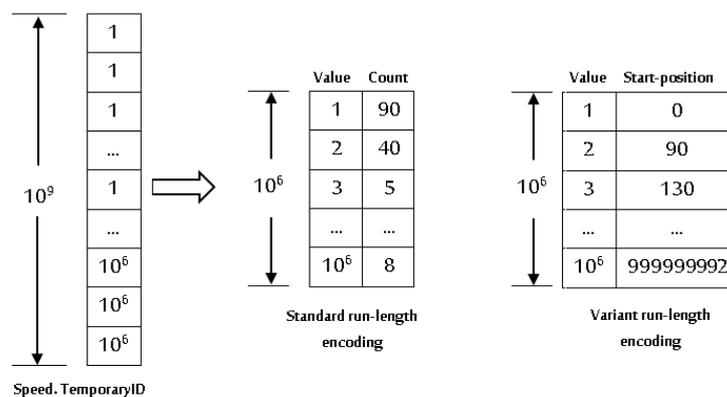
Data Element	Time (Tenths Seconds)																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
MsgCount	D		Not useful vehicle data																	D
TemporaryID	D		Typically unchanged																	D
DSecond	D		Increments by tenths																	D
Latitude, Longitude, Elevation	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Positional Accuracy	D		Typically unchanged																	D
TrasmissionState	D		Typically unchanged																	D
Speed	D		Typically no significant change																	D
Heading	D		Typically no significant change																	D
SteeringWheelAngle	D		Typically no significant change																	D
AccelerationSet4Way	D		Typically no significant change																	D
Brake System Status	D		Typically unchanged																	D
VehicleSize	D		Unchanged																	D

Notes: D indicates new Data.

To apply run-length encoding to a column, the column itself should have the following features: 1. the column is sorted. 2. the fanout of this column is high. The first requirement is easy to understand. If the column is not sorted, then elements with the same values are not grouped together. The second requirement is utilized to measure the average number of duplicates. Only when the number of duplicates is large, run-length encoding can obtain benefits. The definition of fanout is provided in the following.

### 5.2.2. Discussion

Figure 10, gives an example of applying run-length encoding to the (.csv) dataset collected on April 2013, stored into MonetDB database and initially contain more than  $1.9 \times 10^9$  rows.



**Figure 10.** Example of run-length encoding.

The size of the original dataset is 219,990,043,384 bytes (219.99 GB), while the size of the encoded dataset (the standard run-length encoding) is only 43,998,008,676.8 bytes (43.99 GB), which is only 20% of the uncompressed one. Using the run-length encoding data compression method, we succeed in reducing the size of the bandwidth and storage that are required for BSMs by about 80 percent.

### 5.3. Data Delivery

This functionality provides a rich set of APIs to serve not only safety applications, but the different needs of other connected vehicles applications’ (mobility, weathers...). To build up a RESTful Web Services with Spring Boot, Kafka, and Postgres, we first downloaded a Maven project from spring initializer page: (<https://start.spring.io>) shown in Figure 11. Subsequently, we inputted the downloaded project into our Java IDE and commence configuration.

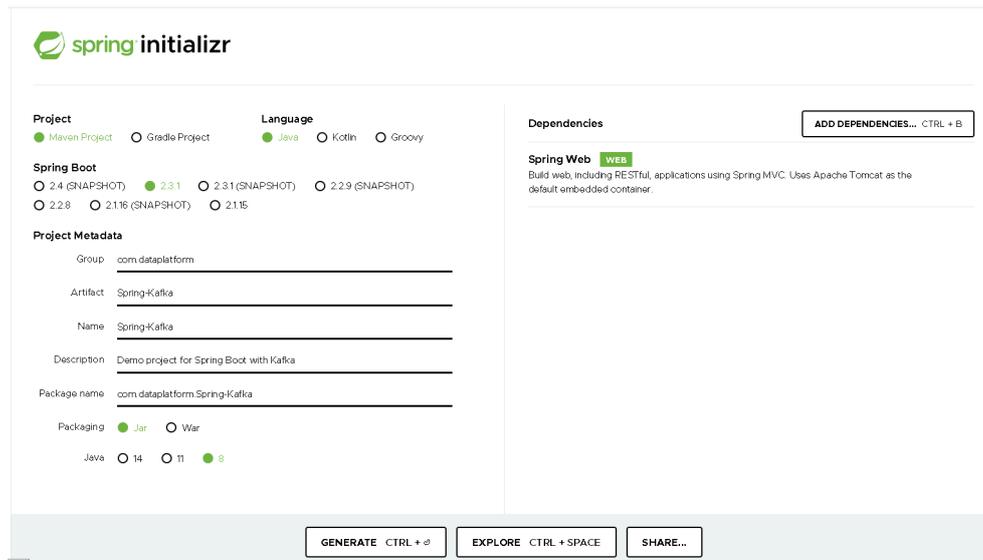


Figure 11. Spring initializer page.

### 5.3.1. Data Sets

We consider a Data Set as a logical storage of related vehicle data elements. Each data set has its own related API endpoints and data elements. The available Data Sets are:

- Points data set: holds kinematic vehicle data. This data consists of data points which are time-stamped vehicle records that contain single or multiple vehicle data elements available like location, speed, etc. points data set are generated from Data Elements of the BSM Part one File.
- Trips data set: contains calculated vehicle trips. An algorithm is used to detect trips from points data. Each trip include details, such as trip start and end times, total trip distance, and location. Also captured in the trip summary file is the distance driven while the vehicle speed was greater than 25 mph. This data element is of interest not only because it further details the trip, but also because it provides a sense of the conditions under which data, for a particular trip, were collected.

The trips data set table contains 11 fields. Table 7, summarizes a list of these fields and a brief description of each where Table 8, provides a few summary measures of the trip data set table from 11 April 2013, and Table 9, provides a 10-calculated trip sample from the trip data set file.

Table 7. Data Elements of the trip data set table.

Field Name	Type
TemporaryID	Integer
TripID	String
Start Date	Date
Start Time	Time
End Date	Date
End Time	Time
Total Trip Distance	Integer
Distance Travelled w/Speed >= 25 mph	Real
Trip Duration	Real
Average Speed	Real
Maximum Speed	Real

**Table 8.** Summary Measures for Data Elements of the trips table.

A	B	C	D	E	F	G	H	I	J	K
10106	297	4/11/2013	7:27:49	4/11/2013	7:36:33	5.12	4.39	524.288	35.18	17.78
10106	300	4/11/2013	16:33:57	4/11/2013	16:42:41	5.83	4.61	524.288	40.07	20.32
10116	716	4/11/2013	8:57:23	4/11/2013	9:08:18	12.99	12.19	655.36	71.37	34.94
10116	718	4/11/2013	12:40:12	4/11/2013	12:42:23	6.43	6.43	131.072	76.68	32.36
10116	719	4/11/2013	14:01:02	4/11/2013	14:03:13	0.64	0.60	131.072	17.77	19.83
10116	720	4/11/2013	22:51:52	4/11/2013	23:02:48	10.48	9.91	655.36	57.60	22.45
10118	771	4/11/2013	8:00:35	4/11/2013	8:15:52	7.42	6.16	917.504	29.12	20.85
10118	772	4/11/2013	14:44:43	4/11/2013	14:49:05	3.31	2.60	262.144	45.48	20.98
10118	773	4/11/2013	17:00:10	4/11/2013	17:06:43	3.98	3.35	393.216	36.44	20.73
10120	671	4/11/2013	7:32:11	4/11/2013	7:51:51	26.37	24.05	1179.648	80.48	35.38

A: TemporaryID; B: TripID; C: Start Date; D: Start Time; E: End Date; F: End Time; G: Total Trip Distance (miles); H: Distance Travelled w/Speed >= 25 mph; I: Trip Duration (seconds); J: Average Speed (m/s); K: Maximum Speed (m/s).

**Table 9.** Sample calculated trip.

Field, Name	Sample, Values	Min., Value	Max., Value	No., of Rows
DeviceID	10204, 10205, 10207	10106	17103	278
TripID	41, 71-A, 50, 167, 87	24	2271	
Start Date	4/11/2013	4/11/2013	4/11/2013	
Start Time	6:52:52, 11:34:40, 13:04:14	0:02:10	23:59:59	
End Date	4/11/2013	4/11/2013	4/11/2013	
End Time	12:40:12, 6:20:06, 11:19:22	0:02:10	23:59:59	
Total Trip Distance	26.816, 29.563, 8.294	0.014572	339.5202	
Distance Trav/Speed >= 25 mph	28.814, 22.377, 1.025	0	333.0411	
Trip Duration	917.504, 786.432, 131.072	131.072	999999	
Avg Speed	0, 1352.017, 3653.507	0.005718	999999	
Max Speed	0, 3.577, 723.406	1.538876	42.25568	

### 5.3.2. Data Delivery Methods

Diverse data consumer use cases may require different data delivery types. For example, an emergency car service may require a real-time event when an accident takes place. On the other hand, Usage-Based Insurance may pull a car’s odometer once a week. Lastly, a data analytics company might opt for historical car data in order to understand traffic trends. Our data platform provides different data delivery methods to cater to these different use case requirements. Table 10 summarizes data delivery methods, where Table 11 summarizes all available historical data APIs.

- Streaming: a ‘push’ mechanism that continuously streams Hot data to a Data Consumer. Streaming uses HTTP POST requests and can send both aggregate and simple data elements. A stream is created by subscribing to a stream. Stream subscription defines one or more data filters such as desired vehicle area (i.e., city), maximal point latency, etc. Streaming is optimal for applications that require real-time, rich, vehicles data.
- Historical data reports: multiple format reports, which contain Cold data. Historical data reports are triggered by a RESTful API call with parameters that define a region (e.g. city) and time span for the report. Report generation may take minutes up to hours to complete. Several historical reports exist for different data elements (e.g., speed, break...).

- Events: an event is defined by a logical rule on one or more data elements. When a rule is set to true, an event message is launched and sent to the data consumer. The system “remembers” that an event has been sent according to a specific rule and will only send it once. Using a Braking pressure data element, an example event maybe a maintenance application which gets an event whenever a vehicle traveling at a certain radius from a maintenance station, crosses a 200-bar braking pressure level (knowing that Higher pressures are not likely. The maximum pressure that the caliper can withstand before breakage is in the range of 250–300 bar. Events are a great way for applications to save processing power and network bandwidth and only get the data they need in real-time.

**Table 10.** Data delivery methods key attributes.

Data Delivery Method	Real Time/Historical	Push/Pull
Streaming	Real-time	Push
Historical Data Reports	Historical	Pull
Events	Real-time	Push

**Table 11.** Available historical data APIs.

API	Request Type	Time Frame	Description
Raw Data	BSM data	Up to 1 month	Provides data points from all the BSMs associated with a specific service
Aggregated Trips Data	Aggregated BSMs Data	Up to 1 month	Provides aggregated data on the various trips vehicles have driven
Trip Points	BSM Data	Length of trip	Provides data points from a specific trip a vehicle has driven

### 5.3.3. Simulation and Discussion

To check out how the API is getting vehicle data behaves, we simulate vehicle trips according to the following steps:

- Step 1: Creating a trip: the first step in our simulating trip data is to configure a route. The starting and ending location may be anywhere. However, our available dataset is limited to Ann Arbor, Michigan, as illustrated in Figure 12. Our trip runs 12.67 miles from the start point at: “S State St, Ann Arbor, MI 48108, USA” to the endpoint at: “M-14, Ann Arbor, MI 48105, USA”. Immediately after we inputted the starting and ending location, we will notice in Figure 12, that the map will provide a visual representation of the driver’s route.
- Step 2: Configure Events: we may select different events to simulate during each test run. The events might occur a couple times within the test but will always occur at least once. At present, we have three available events shown in the Figure 12, bellow.
- Step 3: Run the Simulation: as depicted in Figure 13, once the simulation begins, you will automatically see data coming in every 3 s within the “Point Dataset” tab located on the left side. The vehicle will progress within its route and indicate the location of the driver. Once an event takes place, a small circle will appear on the map. Additionally, the Point dataset timestamp will have a red dot next to it to indicate an event took place at this time. As To see a list of all the events that took place during the simulation, press the “Events” tab located on the left side.

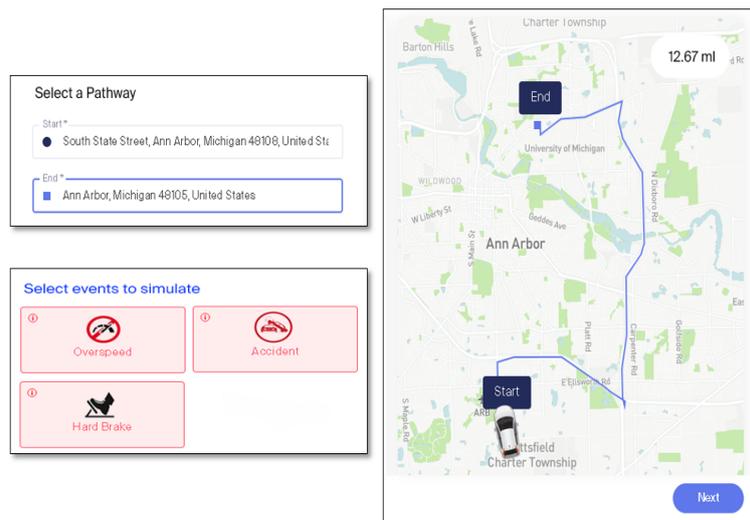


Figure 12. Simulation configuration.

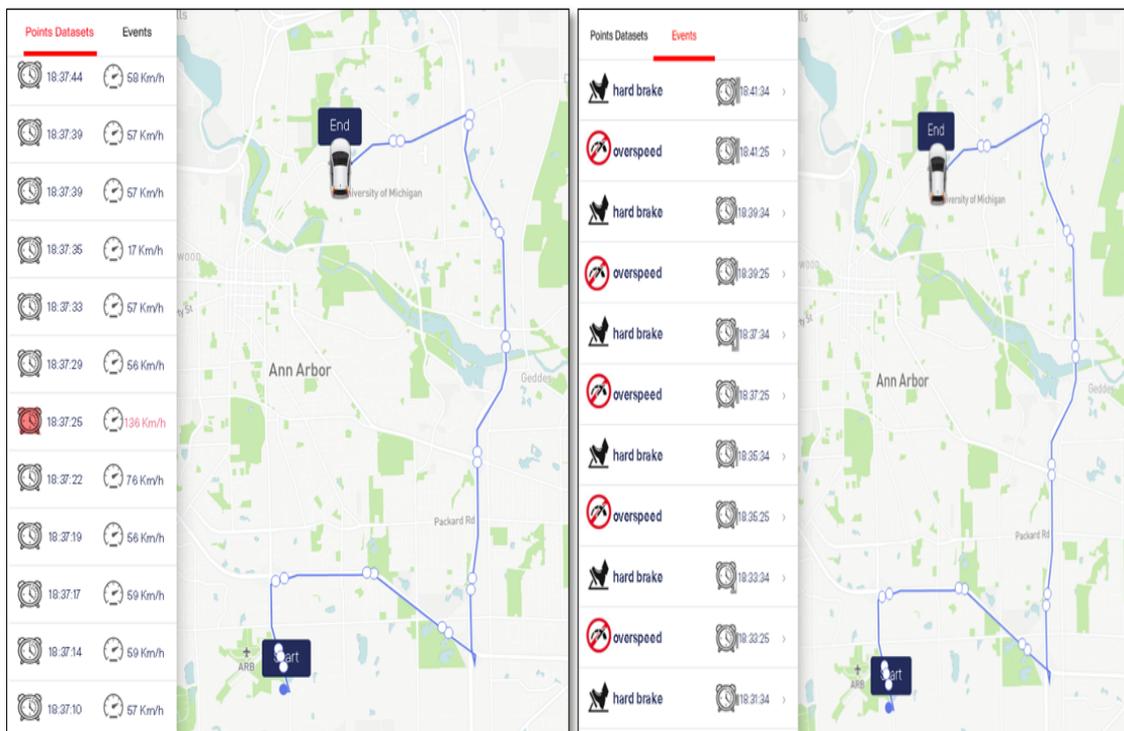


Figure 13. Simulation results.

Previous photos (Figures 12 and 13) show the Importance of maintaining and making BSM data available through multiple delivery methods. Hot data are real-time delivered over a continuous streaming mechanism, while cold data are diffused at regular intervals to increase their value and utility. Undoubtedly, cold data may not be valuable for traffic safety applications as collision avoidance applications, but it may be useful in other applications, such as those that are related to road planning. The following gains can be achieved to name but a few:

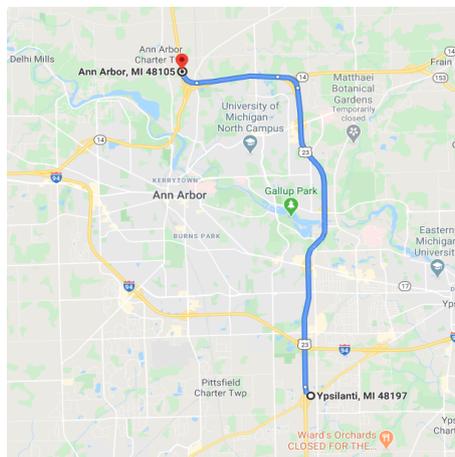
- 1 It gives data consumers the possibility of remotely reuse BSMs data.
- 2 It also allows developers to begin developing their applications without actually having any connected vehicles.
- 3 Real-time traffic information and navigation services and apps can use APIs to highlight areas of congestion and help drivers to find the fastest routes.

### 5.4. Data Forecasting

To take our model a step further, we suggest applying a machine learning (ML) method on our database to perceive whether preserved BSMs can help the road traffic volume prediction (TV). Our study applies the Artificial Neural Network (ANN) approach to predict the traffic volume while using past BSMs data. We deal with the following basic steps of the ML: get the data from database, prepare it, choose a model, train it, evaluate it, export it, and make the predictions available for use. Our development environment has relied on TensorFlow as the framework. For the generating part, we made use of Python, with Jupyter Notebooks, and for the prediction serving part, we adopted Java, while using Spring Boot.

#### 5.4.1. Data Extraction

The SPM model deployment was conducted in Ann Arbor, Michigan. The field test includes 75 miles of instrumented roadway. Our dataset sample covered the amount of five days traffic activity that occurred over 6 h, during the period 6:00 a.m. to 12:00 p.m. (including congested and smooth traffic regimes); across a distance of 11.2 miles, as shown in Figure 14.



**Figure 14.** Road segment used for traffic volume prediction (from: Ypsilanti, Michigan 48197, USA to Ann Arbor, Michigan 48105, USA).

Vehicles that participated in this study include light/passenger vehicles, heavy/commercial trucks, and busses. Based on DE\_VehicleType data elements, vehicles were classified in three main categories: Passenger car, Bus, and Trailer.

*Traffic volume can be determined by counting the number of vehicles that cross through a point on a road segment at a specific time and denoted by vehicles per hour (v/h).*

We manually calculated the TV using the Position data frame by counting the number of vehicles traversing the road segment illustrated in Figure 14. It is noteworthy that we did not examine the Heading data element because we took into account both directions. Data extraction displayed in Table 12, was performed in 5 min. intervals.

**Table 12.** Five minutes of speed and traffic volume.

Type of Vehicle	Both Directions							
	Average Speed (mph)				Volume			
	Minimum	Maximum	MEAN	SD	Minimum	Maximum	MEAN	SD
Passenger car	44.4280	48.4669	46.6463	1.1060	38.0000	60.0000	46.2540	4.3870
Bus	30.7579	32.6220	31.7148	0.8326	4.0000	24.0000	10.5500	3.3250
Trailer	18.3304	19.8839	18.7716	1.1744	3.0000	33.0000	11.4100	6.0900

#### 5.4.2. Model Development

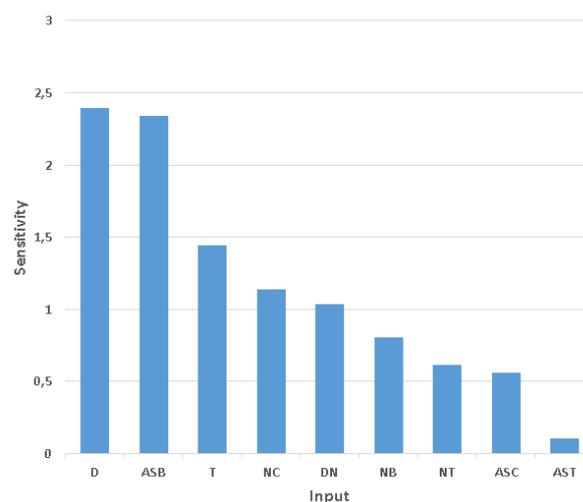
We performed the traffic volume prediction using the Multilayer Perceptron (MLP) artificial neural network which is a feedforward ANN. This kind of ANN relies on backpropagation for the training stage. It has multiple input layers connected as a directed graph with output layers. Analyzed dataset contains the following features shown in Table 13: 1-Date, 2-Time, 3-Number of Passenger car, 4-Number of Bus, 5-Number of Trailer, 6-Average speed of Passenger car, 7-Average speed of Bus, 8-Average speed of Trailer and 9-Traffic density. The whole inputs are significant except for the AST input, as is shown in Table 13 and is observable in Figure 15.

*Generally measured in units of vehicles per mile (v/m), the traffic density is referred to the volume of vehicles on a road fragment.*

**Table 13.** Input variables considered by the Artificial Neural Network (ANN).

N°	Features	Acronym	Sensitivity
1	Date	D	3986
2	Time	T	1.4418
3	NB Passenger car	NC	1.1359
4	NB Bus	NB	0.8068
5	NB Trailer	NT	0.7001
6	AVG speed of Passenger car	ASC	0.5584
7	AVG speed of Bus	ASB	2.4001
8	AVG speed of Trailer	AST	0.1083
9	Traffic density	DN	1.0385

As a preprocessing step, we first randomized our dataset and then divided it into three sub-sets. The first sub-set represented 10% of the whole dataset and it was used for training. The second sub-set was taken for cross-validation in the ratio of 10%. The remaining part that represent 80% of the dataset was used for testing purposes.



**Figure 15.** Input variables sensitivity analysis.

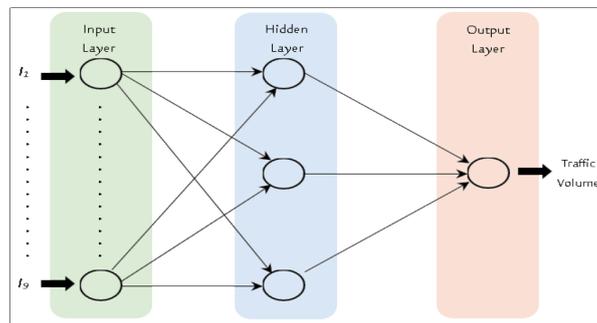
Distinct architectures of ANN have been designed to determine an efficient network. Table 14, presents several ANN models that were prepared to train on the dataset. The testing data sets along

with cross-validation were used to identify the performance of every ANN architecture. As well as, to evaluate the predicted results, other parameters were exploited, such as Normalized Mean Square Error (NMSE), the coefficient of correlation (*r*), and Mean Absolute Error (MAE). The desired neural network is that of three hidden neurons, according to Table 14. Therefore, Figure 16 shows, in detail, the ANN structure used in our work.

**Table 14.** Several ANN architectures to predict TV.

A	B	C	D	E	F	G	Training		Cross-Val		Testing			
							Min MSE (*103)	Final MSE (*103)	Min MSE (*103)	Final MSE (*103)	MSE	NMSE (*102)	MAE (*10)	r (%)
A1	Sigmoid	1	4	200	Mom	1/0.7	13.10	13.10	19.46	19.46	223.13	69.88	109.97	84.42
A2	Sigmoid	1	5	300	Mom	1/0.7	7.80	7.80	13.85	-	147.24	48.71	82.77	89.82
A3	Sigmoid	1	7	500	Mom	1/0.7	2.90	2.90	4.10	4.10	55.95	17.46	49.57	97.02
A4	Tanh Axon	1	6	700	Mom	1/0.7	0.35	0.35	0.50	0.50	2.70	0.68	12.50	99.62
A5	Tanh Axon	1	3	300	Lev	-	0.16	0.17	0.15	0.15	0.74	0.22	6.28	<b>99.88</b>
A6	Tanh Axon	1	4	400	Lev	-	0.60	0.60	0.81	0.90	2.16	0.69	13.00	99.64

A—Architecture; B—Activation function; C,D—Number of hidden layer and neurons respectively; E—Epoch; F—Algorithm Learning; G—Adaptive step size/Mom; Mom—Momentum; Lev—Levenberg; r—the correlation coefficient.



**Figure 16.** Architecture of the desired ANN.

5.4.3. Results and Discussion

The linear correlation between the predicted and real volume values that are illustrated in Figure 17, as well as the output summary of the qualified ANN test stage presented in Table 15, confirm that our developed model gives accurate results. The scatterplot that is shown in Figure 17 determines the solid relationship between predicted traffic volume (Y) and real traffic volume (X) confirmed by the correlation coefficient value very close to 1. This can be explained by the fact that the predicted traffic volume values are highly fitted to the real traffic volume values.

**Table 15.** Output summary of the qualified ANN test stage.

SD	RMES	r	MAE	X2 Calculated	X2 Tabulated
0.8713	0.8726	0.9988	0.7011	5.0102	143.164

Bearing in mind that our primary goal one hand and considering the good value of the coefficient of correlation and very small errors, on the other hand, it can be argued that our developed ANN has successfully reused stored BSM as past data during training, cross-validation, and testing stage to

accurately predict future traffic volume despite for a short time. Thus, stored BSM can serve as a rich dataset for machine learning in the transportation field.

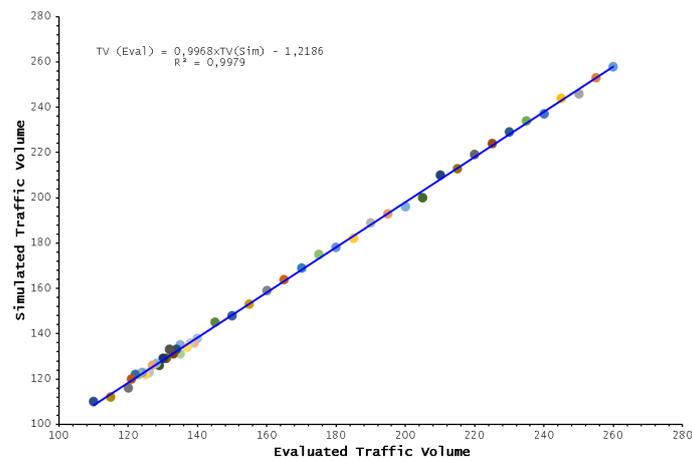


Figure 17. Predicted traffic volume vs Real traffic volume.

### 5.5. Data Visualization

Data visualization involves visually displaying information to present a point or perspective on specific data. Making the development of a web-based visualization platform easier and faster pushes us to choose some suitable development frameworks. Adopting the design philosophy of Model-View-Controller (MVC), we set up our data monitoring service by the integration of MyBatis at the Data Access Object (DAO) level and Echart 4.8 as the front-end visualization controller with Springboot.

Echart is an open-sourced, web-based, cross-platform framework that has a powerful function, friendly interface, excellent performance to enhance data visualization. This excellent development tool for front-end developers can help to present huge amounts of data to users in a very appropriate way, and users can analyze valuable information through charts. Using Echart, we can visualize a bunch of information based on data elements and data frames or according to time and geofencing technique.

BSMs are stored according to data elements and Data frames data, as shown in Figure 7. An overview of different elements visualization is shown in Figures 18a,b and 19a,b). This element visualization provides a ready means to tell stories from the atomic data as well as provides us with analysis at various levels of detail. The red color in Figure 18b gives an idea about speedy vehicles over time in multiple geographic areas, which allow executives to drill down into specific locations to see what is being done well or poorly.

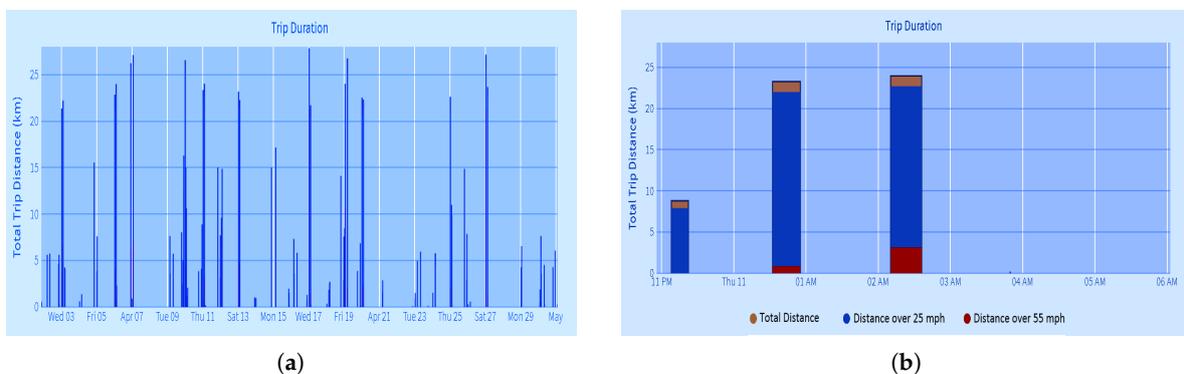
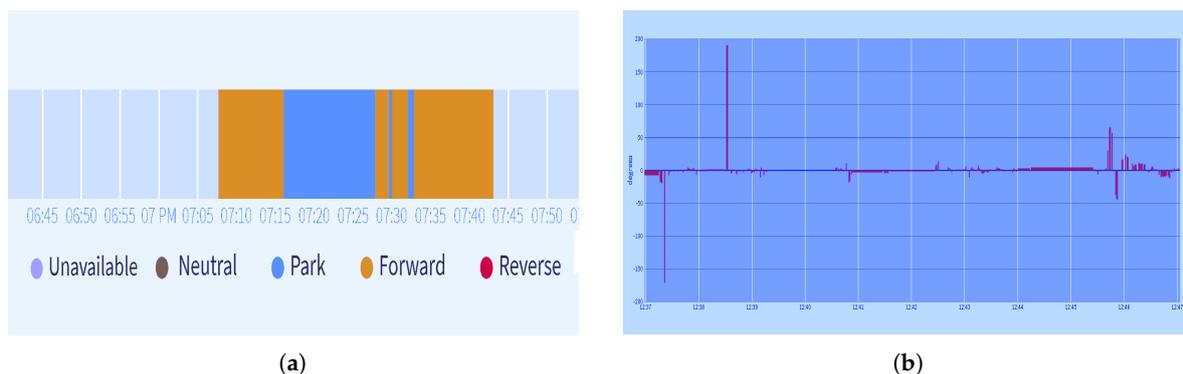


Figure 18. Visualization of the Trip Summary table content. (a) BSM Trip Summary (full time span = 29 days); (b) BSM Trip Summary Zoomed In (4/11/2013).



**Figure 19.** Visualization of Data Elements. (a) Transmission State DE; (b) Steering Wheel DE.

## 6. Conclusions

Given our primary goal of not constraining BSM use contextually and geographically, in this paper, we introduced a new philosophy that aims at conserving collected BSMs and adopting the In-Vehicle Computing paradigm in order to create a reliable and useful transport data source. We then proposed our new BSM reuse model based on three-stage process. In its first stage, our proposed model captures generated and acquired BSMs. Then, in the second stage, it would perform a series of procedures on the raw BSMs to be storable according to the proposed model. In the third stage, our model aims at opening up new possibilities for the endless reuse of stored BSMs.

Later in our study case, we built an embedded data platform accrediting the Model-View-Controller design commonly used for developing user interfaces. This new platform has accomplished several purposes of data reduction, delivery, and visualization. We were able to perform lossless data compression and considerably reduce the data size; the thing that has a positive impact on bandwidth and storage requirements that have been reduced by about 80%. We have also achieved different data delivery according to the Pull and Push mechanisms to cater to the different data consumer use case requirements. Adopting the ANN paradigm, we obtained an accuracy of 0.9988 in carrying out traffic volume prediction. We attained the visualization of some data elements to enhance analytics and support decisions-making for transportation.

Our work bears certain limitations that should be recognized. First, the proposed model was partially developed using a pre-collected BSMs, forthcoming works should focus on the data capture stage. Second, our in-vehicle platform has been restricted to an isolated vehicle far from real-world traffic, so the number of vehicles, as well as real-world traffic conditions, should be considered in future work. Third, our in-vehicle platform was tested in a private wide area network by limited users, further works are required to correct this deficiency by making our platform publicly available.

**Author Contributions:** K.B. was involved in all parts of the study, including conceptualization, methodology, software, investigation, visualization, original draft preparation, reviewing, and editing. S.B. was mainly involved in conceptualization, interpretation, and discussion of results, reviewing, and editing. A.M. was mainly involved in supervision and validation. All authors have read and agreed to the published version of the manuscript

**Funding:** This research was funded by the University of MEDEA with the LESIA laboratory of the University of Biskra.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results

## References

1. U.S. DOT. Recommended Practices for DSRC Licensing and Spectrum Management. 2015. Available online: [https://rosap.ntl.bts.gov/view/dot/3577/dot\\_3577\\_DS1.pdf](https://rosap.ntl.bts.gov/view/dot/3577/dot_3577_DS1.pdf) (accessed on 27 July 2020).

2. Vehicle Safety Communications—Applications, Technical Report, U.S.DOT. 2011. Available online: <https://www.nhtsa.gov/sites/nhtsa.dot.gov/files/811492a.pdf> (accessed on 27 July 2020).
3. Kenney, J.B. Dedicated Short-Range Communications (DSRC) Standards in the United States. *Proc. IEEE* **2011**, *99*, 7. [[CrossRef](#)]
4. MacHardy, Z.; Khan, A.; Obana, K.; Iwashina, S. V2X Access Technologies: Regulation, Research, and Remaining Challenges. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 1858–1877. [[CrossRef](#)]
5. Benaissa, K.; Bitam, S.; Mellouk, A. Efficient Messages Broadcasting within Vehicular Safety Applications. In Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, Canada, 8–13 October 2017.
6. Benaissa, K.; Bitam, S.; Mellouk, A. Context-based BSM aggregation for broad-scale applications in vehicular networks. In Proceedings of the 2018 IEEE 43rd Conference on Local Computer Networks (LCN), Chicago, IL, USA, 1–4 October 2018.
7. Miucic, R. (Ed.) *Connected Vehicles Intelligent Transportation Systems*; Springer: Basel, Switzerland, 2019.
8. Siegel, J.E.; Erb, D.C.; Sarma, S.E. A Survey of the Connected Vehicle Landscape, Architectures, Enabling Technologies, Applications, and Development Areas. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 2391–2406. [[CrossRef](#)]
9. Ali, R.Y.; Gunturi, V.M.V.; Shekhar, S.; Eldawy, A.; Mokbel, M.F.; Kotz, A.J.; Northrop, W.F. Future Connected Vehicles: Challenges and Opportunities for Spatio-temporal Computing. In Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, WA, USA, 3–6 November 2015.
10. Khan, J.A.; Doudane, Y.G. ROVERS: Incentive-Based Recruitment of Connected Vehicles for Urban Big Data Collection. *IEEE Trans. Vehic. Tech.* **2019**, *68*, 5281–5294. [[CrossRef](#)]
11. Abdelhamid, S.; Hassanein, H.S.; Takahara, G. Reputation-aware, trajectory-based recruitment of smart vehicles for public sensing. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 1387–1400. [[CrossRef](#)]
12. Jalooli, A.; Song, M.; Wang, W. Message coverage maximization in infrastructure-based urban vehicular networks. *Veh. Commun.* **2019**, *16*, 1–14. [[CrossRef](#)]
13. Liu, K.; Ng, J.K.; Lee, V.; Son, S.H.; Stojmenovic, I. Cooperative data scheduling in hybrid vehicular ad hoc networks: VANET as a software defined network. *IEEE/ACM. Trans. Netw.* **2016**, *24*, 1759–1773. [[CrossRef](#)]
14. Du, Y.; Chowdhury, M.; Rahman, M.; Dey, K.; Apon, A.; Luckow, A.; Ngo, L.B. A Distributed Message Delivery Infrastructure for Connected Vehicle Technology Applications. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 787–801. [[CrossRef](#)]
15. Si, P.; He, Y.; Yao, H.; Yang, R.; Zhang, Y. DaVe: Offloading Delay-Tolerant Data Traffic to Connected Vehicle Networks. *IEEE Trans. Veh. Technol.* **2016**, *65*, 3941–3953. [[CrossRef](#)]
16. Luan, T.H.; Cai, L.X.; Chen, J.; Shen, X.; Bai, F. Engineering a Distributed Infrastructure for Large-Scale Cost-Effective Content Dissemination over Urban Vehicular Networks. *IEEE Trans. Veh. Technol.* **2014**, *63*, 1419–1435. [[CrossRef](#)]
17. Zeng, L.; Zhang, J.; Han, Q.; Ye, L.; He, Q.; Zhang, X.; Yang, T. A Bus-Oriented Mobile FCNs Infrastructure and Intra-Cluster BSM Transmission Mechanism. *IEEE Access* **2019**, *7*, 24308–24320. [[CrossRef](#)]
18. Uppoor, S.; Trullols-Cruces, O.; Fiore, M.; Barcelo-Ordinas, J.M. Generation and Analysis of a Large-Scale Urban Vehicular Mobility Dataset. *IEEE Trans. Mob. Comput.* **2014**, *13*, 1061–1075. [[CrossRef](#)]
19. Gerla, M.; Wu, C.; Pau, G.; Zhu, X. Content distribution in VANETs. *Veh. Commun.* **2014**, *1*, 3–12. [[CrossRef](#)]
20. Abboud, K.; Omar, H.A.; Zhuang, W. Interworking of DSRC and cellular network technologies for V2X communications: A survey. *IEEE Trans. Veh. Technol.* **2016**, *65*, 9457–9470. [[CrossRef](#)]
21. Gu, Y.; Cai, L.X.; Pan, M.; Song, L.; Han, Z. Exploiting the stable fixture matching game for content sharing in D2D-based LTE-V2X communications. In Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016.
22. *Connected Car Industry Report*; Spain Telefonica Machina Research Report; 2013. Available online: <https://machinaresearch.com/news/telefonica-in-conjunction-with-machina-research-publishes-connected-car-industry-2013-report/> (accessed on 27 July 2020).
23. Wright, J. Intelligent Transportation Systems: Vehicle-to-Infrastructure Technologies Expected to Offer Benefits, but Deployment Challenges Exist. 2015. Available online: <https://www.gao.gov/assets/680/672548.pdf> (accessed on 27 July 2020).

24. Dennis, E.P.; Spulber, A. International scan of connected vehicle technology deployment efforts. *Auto. Cent. Res. Tech. Rep.* **2017**.
25. Rettore, P.H.; Maia, G.; Villas, L.A.; Loureiro, A.A.F. Vehicular Data Space: The Data Point of View. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 3. [[CrossRef](#)]
26. Campolo, C.; Molinaro, A.; Scopigno, R. Vehicular ad hoc Networks. In *Standards, Solutions, Research*; Springer: Cham, Switzerland, 2015; pp. 83–122.
27. Dedicated Short Range Communications Message Set Dictionary. *SAE Inter. Tech Rep.* **2016**. Available online: [https://www.sae.org/standards/content/j2735\\_200911/](https://www.sae.org/standards/content/j2735_200911/) (accessed on 27 July 2020).
28. D. J. Abadi, S. R. Madden, and N. Hachem. Column-stores vs. row-stores: How different are they really? In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*; Association for Computing Machinery: New York, NY, USA, 2008; pp. 967–980.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).